Security Implications of Permission Models in Smart-Home Application Frameworks

ABSTRACT

Smart-home technology has evolved beyond basic convenience functionality, such as automatically controlled lights and door openers, to provide tangible benefits. For instance, water flow sensors and smart meters facilitate energy efficiency. IP-enabled cameras, motion sensors, and connected door locks offer better control of home security. However, attackers can manipulate smart devices to cause users physical, financial, and harm. For example, burglars can target a connected door lock to plant hidden access codes. Early smart-home systems had steep learning curves and complicated device setup procedures and thus were limited to do-it-yourself enthusiasts. (Many forums exist for people to exchange know-how, such as forum universal-devices.com.)

EXISTING SYSTEM

In Existing System, smart-home systems had steep learning curves and complicated device setup procedures and thus were limited to do-it-yourself enthusiasts. Recently several companies introduced cloud-backed systems that are easier for users to set up and that provide a programming frame-work for third party developers to build smart-home apps. For Example, the auto-lock Smart App, available in the Smart Things app store, requires only the lock command of capability lock but also gets access to the unlock command, thus increasing the attack surface if the Smart App is exploited. If the lock command is misused, the Smart App could lock out authorized household members, causing inconvenience; however, if the unlock command is misused.

DIS ADVANTAGES

- Increasing the attack surface if the Smart App is exploited.
- Stealing a device identifier and generating fake carbon monoxide sensor readings.

PROPOSED SYSTEM

In Proposed System, using four concrete attacks, we demonstrated how the over privilege design issue weakens home security. We combined over privilege with other security design flaws in the Smart Things framework to make the attacks remote and stealthy. The backdoor PIN code injection attack uses coarse Smart App–Smart Device binding over privilege to force an existing Smart App to program a PIN code into a door lock. The door lock PIN code snooping attack is a stealthy malware app that uses over privilege in the event system of Smart Things to snoop on PIN codes as they're created and then leak them out.

ADVANTAGES

- Reduce automatic over privilege and better balance capability granularity and usability.
- The device identifiers are easy to exchange among Smart Apps.

SYSTEM REQUIREMENTS

H/W System Configuration:-

Processor	-	Pentium –III
RAM	-	256 MB (min)
Hard Disk	-	20 GB
Key Board	-	Standard Windows Keyboard
Mouse	-	Two or Three Button Mouse
Monitor	-	SVGA

S/W System Configuration:-

Operating System	:	Windows95/98/2000/XP
Application Server	:	Tomcat5.0/6.X
Front End	:	HTML, Jsp

Scripts	:	JavaScript.
Server side Script	:	Java Server Pages.
Database	:	MySQL 5.0
Database Connectivity	:	JDBC